

Testing Like A Boss

Arkady Galyash

TosChart

April 29, 2014



Agenda

1. How regular JUnit test looks like (in time)?
2. Links



Intro

- John Doe The Programmer
- Java developer @ Moon Ms
- Since 2000
- Binary search algorithm



June 2000



Just graduated, "tests are for chicken"



May 2001



JUnit discovered



July 2006



User tries to search non-existent element



December 2007



User tries to search in array with duplicates



October 2009



Integer overflow



Long way



2000



2001



2006



2007



2009

xUnit

- We need to think about test cases
- Until we've discover one, we cannot find a bug
- If we discover a test case, this test is useless right now, it may be useful only in future



QuickCheck

- Another approach
- You specify pre- and post-conditions
- Let computer generate input data

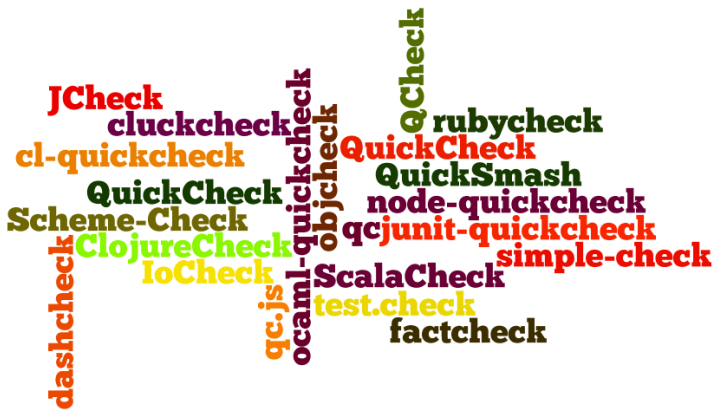


QuickCheck History

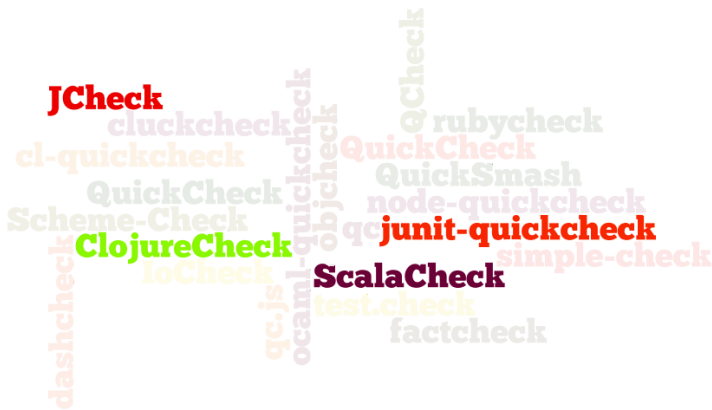
- Initially written for Haskell in 1999
- Re-implementations of QuickCheck exist for Java, Scala, Clojure, Groovy, ...



QuickCheck implementations



QuickCheck implementations





Demo



Links

- Introduction to QuickCheck2



Thank You!

